Tecnológico de Monterrey

Análisis de Grandes Volúmenes de Datos

Avance de Proyecto 4:

Sistema de Recomendación

Equipo 39

Luis Gerardo Barbosa Mendoza | A01731203

Miguel Ángel Marines Olvera | A01705317

Genaro Rodríguez Vázquez | A01150931

Luis Ángel Seda Marcos | A01795301

16 / 06 / 2024

1. Implementación

- Implementación Sistema de Recomendación 1

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 1): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Este sistema de recomendación de películas utiliza una red neuronal autoencoder para hacer recomendaciones personalizadas en base a los usuarios, las películas vistas y la calificación de las películas.

  Para implementar el sistema se hace lo siguiente:

  1. Creación de la Tabla Pivote: Se crea una tabla pivote donde cada fila representa a un usuario y cada columna representa una película, con las calificaciones dadas por los usuarios. Las calificaciones faltantes se llenan con ceros.

  2. Normalización de Datos: Se normalizan las calificaciones para que estén en un rango de 0 a 1, lo cual mejora la eficiencia del entrenamiento del modelo.

  3. División de Datos: Se dividen los datos en conjuntos de entrenamiento y prueba, usando el 80% de los datos para entrenamiento y el 20% para prueba.

  4. Definición y Construcción del Modelo: Se define y construye un modelo de red neuronal (autoencoder) con:

     - Una capa de entrada que toma como entrada el número de películas.

     - Una capa oculta densa con 256 nodos y función de activación sigmoid.

     - Una capa de salida con un nodo por cada película.

     - El modelo se compila usando el optimizador Adam y la pérdida se mide con el error cuadrático medio (MSE). También se añaden las métricas adicionales MAE y MAPE.

  5. Entrenamiento del Modelo: Se entrena el modelo usando los datos de entrenamiento, configurado para 50 épocas y un tamaño de lote de 64. Dado que se trata de un autoencoder, tanto la entrada como el objetivo son los mismos datos.

  6. Evaluación del Modelo: Se evalúa el modelo en el conjunto de prueba, imprimiendo la pérdida y las métricas adicionales (MAE, MSE, MAPE).

  7. Cálculo del RMSE: Se calcula el Root Mean Squared Error (RMSE) para tener una métrica adicional de evaluación del modelo.

  8. Visualización de la Historia de Entrenamiento: Se grafica la pérdida de entrenamiento y validación a lo largo de las épocas para visualizar el desempeño del modelo durante el entrenamiento.

  10. Al final, se realiza una prueba dando como input el número de un usuario X y se obtiene el top 10 de recomendaciones de películas.

Como referencia se utilizó el usuario con el ID número “10”, dónde el algoritmo devolvió las respectivas recomendaciones acorde a las características.

```
#Intentar con diferentes user_ids para obtener las peliculas recomendadas
get_Recommendations_for_user(10)
```

```
1/1 [==============================] - 0s 87ms/step
Top 10 movie recommendations for the sample user:
title: Secret of Roan Inish, The (1994), Predicted Rating: 0.5054865479469299
title: Hate (Haine, La) (1995), Predicted Rating: 0.44272491335868835
title: Miracle on 34th Street (1994), Predicted Rating: 0.4089670777320862
title: Just Cause (1995), Predicted Rating: 0.38791871070861816
title: Lost Weekend, The (1945), Predicted Rating: 0.37184029817581177
title: Poetic Justice (1993), Predicted Rating: 0.3688317537307739
title: Singin' in the Rain (1952), Predicted Rating: 0.3478822410106659
title: Go Fish (1994), Predicted Rating: 0.3141983151435852
title: Some Like It Hot (1959), Predicted Rating: 0.3121604025363922
title: Being Human (1993), Predicted Rating: 0.3114105463027954
```

- Implementación Sistema de Recomendación 2

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 2): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Este sistema de recomendación de películas utiliza las palabras clave, el elenco, el director, los géneros, el título, la sinopsis y la compañía de producción como características para calcular la similitud entre películas y recomendar películas similares a una película de entrada específica y hacer recomendaciones personalizadas.

  Este sistema de recomendación de películas implementa técnicas de procesamiento de texto, análisis de similitud y modelado de datos.

  Para implementar el sistema se hace lo siguiente:

  1. Creación de una Colección Unificada de Características

  Al conjunto de datos se agrega una columna adicional llamada “collection” en la que se concatenan todas las características a tomar en cuenta (palabras clave, elenco, director, géneros, título, sinópsis y compañía de producción) en una sola cadena de texto por película.

  Esta colección unificada facilita el procesamiento, ya que consolida toda la información relevante de una película en un solo campo.

  2. Transformación de Texto

  Se eliminan los espacios en blanco y se convierte todo a minúsculas.

  Se aplica el algoritmo de stemming, “PorterStemmer”, de la librería “nltk”, que reduce las palabras a sus raíces morfológicas, lo cual es crucial para manejar las variantes de las palabras y mejorar la coincidencia de términos.

  3. Vectorización de Texto

Se utiliza “CountVectorizer” de la librería “scikit-learn” para convertir las cadenas de texto en una representación numérica que sea más fácil de procesar. Se configuran con un máximo de 5000 características y se eliminan las “stop words”.

La salida de “CountVectorizer” es una matriz dispersa que representa la frecuencia de cada término en el corpus.

4. Cálculo de Similitud - Similitud Coseno

La recomendación de sistema por similitud de coseno es un enfoque utilizado en sistemas de recomendación para calcular la similitud entre dos elementos basándose en sus características.

El cálculo de la similitud de coseno se basa en el concepto de espacio vectorial, donde cada elemento se representa como un vector en un espacio multidimensional, donde cada dimensión corresponde a una característica.

El algoritmo de similitud de coseno mide el ángulo entre dos vectores, lo que refleja la similitud direccional entre ellos. Cuanto más cercano estén los vectores en dirección (es decir, cuanto menor sea el ángulo entre ellos), mayor será la similitud de coseno y, por lo tanto, mayor será la similitud entre los elementos.

Para calcular la similitud de coseno entre dos elementos, se utiliza la siguiente fórmula:

$$\text{similitud}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \cdot \|\mathbf{B}\|}$$

Donde:

- $\mathbf{A} \cdot \mathbf{B}$ es el producto punto entre los vectores $\mathbf{A}$ y $\mathbf{B}$.
- $\|\mathbf{A}\|$ y $\|\mathbf{B}\|$ son las magnitudes (normas) de los vectores $\mathbf{A}$ y $\mathbf{B}$ respectivamente.

En este contexto, la similitud de coseno ayuda a determinar qué tan similares son las películas en función de sus descripciones textuales vectorizadas.

5. Algoritmo de Recomendación

Se implementa una función de recomendación dónde:

1. Se toma el título de una película como entrada y se busca su índice en el DataFrame.
2. Se calculan las distancias de similitud coseno entre la película dada y todas las demás películas.
3. Se ordenan las distancias en orden descendente y se seleccionan las cinco películas más similares.
4. Se muestran los títulos de las películas recomendadas.

Como referencia se utilizó la película “Indiana Jones and the Last Crusade”, dónde el algoritmo devolvió las respectivas recomendaciones acorde a las características.

```
[ ]  1 recommend('Indiana Jones and the Last Crusade')
```

```
Raiders of the Lost Ark
Indiana Jones and the Temple of Doom
Lara Croft: Tomb Raider
King Solomon's Mines
National Treasure
```

2. Evaluación

- Evaluación Sistema de Recomendación 1

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 1): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Las métricas a utilizar para evaluar el desempeño de este sistema son las siguientes:

RMSE (Root Mean Squared Error) - Error Cuadrático Medio de la Raíz:

- El RMSE mide la diferencia entre los valores predichos por el modelo y los valores reales de las calificaciones. Es útil para evaluar la precisión de las predicciones.
- Fórmula: $\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$

  donde y_i son las calificaciones reales y $\hat{y}_i$ son las calificaciones predichas.

MAE (Mean Absolute Error) - Error Absoluto Medio:

- El MAE mide la magnitud promedio de los errores en un conjunto de predicciones, sin considerar su dirección. Es más interpretativo en términos de unidades de las calificaciones.
- Fórmula: $\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$

MAPE (Mean Absolute Percentage Error)

- **Definición:** Mide la precisión de un modelo de predicción en términos porcentuales.
- **Interpretación:** Proporciona una idea clara de la precisión relativa del modelo comparada con los valores reales, expresada como un porcentaje.
- **Ventajas:**
  - Independiente de la escala de los datos.
  - Fácil de interpretar.
- **Desventajas:**
  - Sensible a valores reales cercanos a cero, lo que puede causar valores muy altos o indefinidos.

Fórmula del MAPE

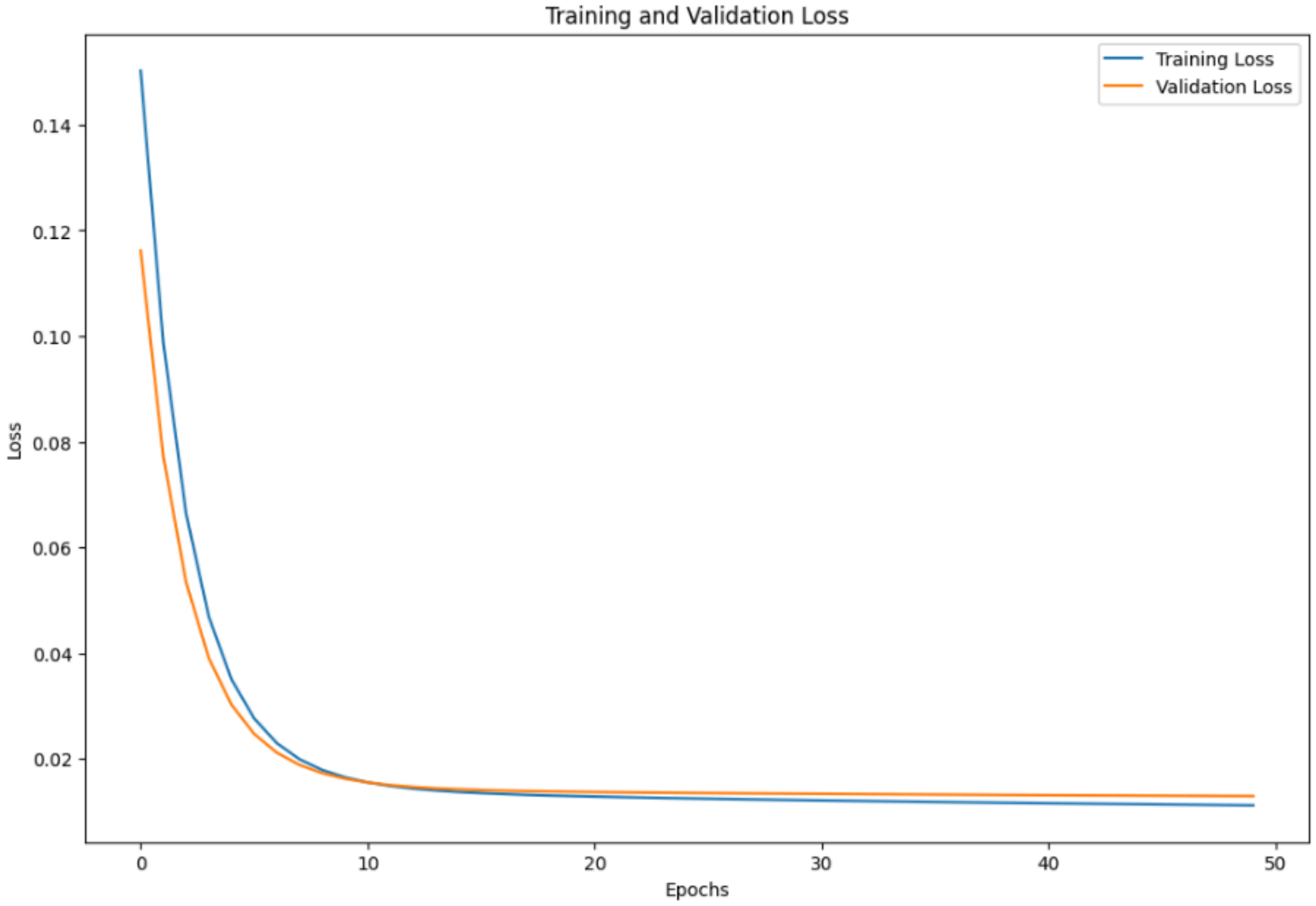
$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$

donde:

- y_i son las calificaciones reales.
- $\hat{y}_i$ son las calificaciones predichas.
- n es el número de observaciones.

Resultados con 50 epocas:

```
Root Mean Squared Error (RMSE): 0.11379894263401093
4/4 - 0s - loss: 0.0130 - mae: 0.0594 - mse: 0.0130 - mape: 47885120.0000 - 48ms/epoch - 12ms/step
Loss: 0.012950199656188488
Mean Absolute Error (MAE): 0.05944839119911194
Mean Squared Error (MSE): 0.012950199656188488
Mean Absolute Percentage Error (MAPE): 47885120.0
```



- Evaluación Sistema de Recomendación 2

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 2): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  1. Evaluación con Base a la Similitud de Coseno

  En este sistema de recomendación basado en la similitud por coseno, donde se utilizan características de palabras clave asociadas a cada película como los géneros, las sinopsis, las compañías de producción, el elenco y los directores, un valor de similitud mayor a 0.20 representa un muy buen desempeño debido a la naturaleza dispersa y multidimensional de los datos. En este contexto, las características que describen cada película son numerosas y variadas, lo que resulta en vectores de alta dimensionalidad. Dado que la similitud por coseno mide el ángulo entre estos vectores, un valor superior a 0.20 indica que dos películas tienen un grado considerable de similitud en varios de estos aspectos, a pesar de la diversidad de datos. Esto es especialmente significativo porque, en un espacio vectorial tan amplio, la mayoría de los pares de películas tienden a tener valores de similitud bajos, cercanos a 0. Por tanto, alcanzar un valor superior a 0.20 sugiere que las películas comparten suficientes características relevantes para que la recomendación sea efectiva y precisa.

En el sistema se logra obtener un valor mayor a 0.20, en las 5 recomendaciones que se hacen, lo cual índica que el sistema tiene un buen desempeño.

2. Evaluación de Correlación de las Películas Recomendadas en Base a su Similitud de Coseno.

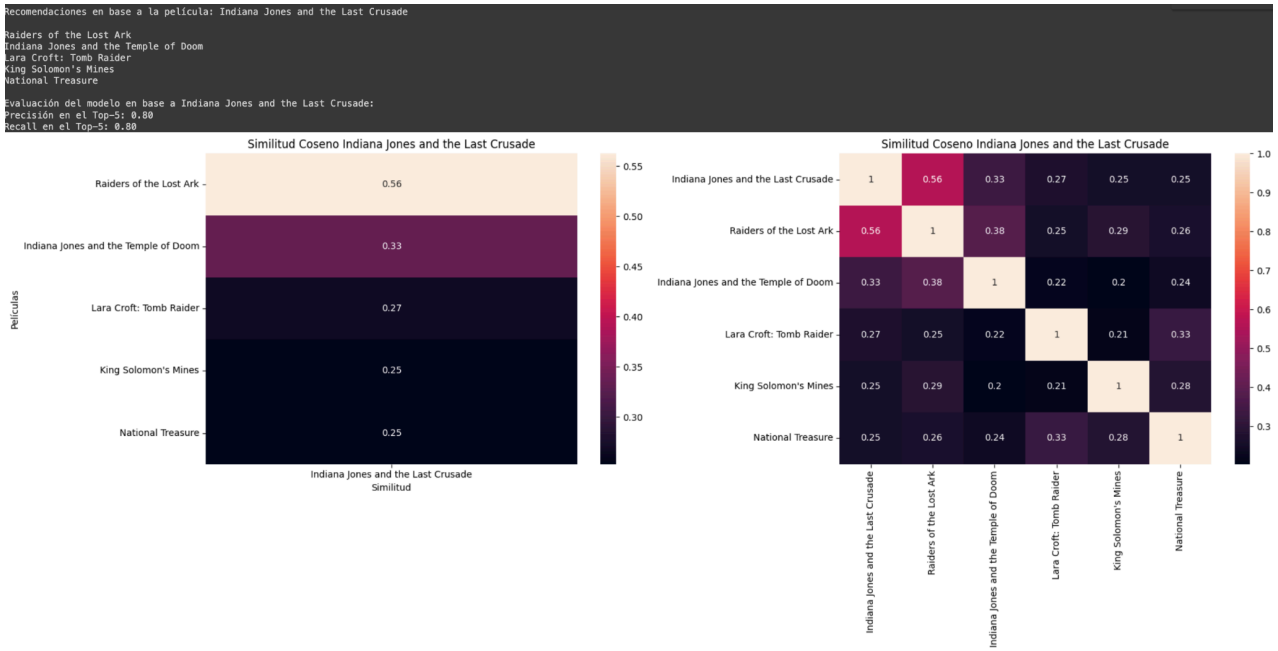
De igual manera, se logra obtener una correlación mayor a 0.20 en la similitud de coseno entre las 5 películas recomendadas. Esto indica que el sistema tiene un buen desempeño.

3. Evaluación de Precisión y Recall en Base a Otros Sistemas de Recomendación.

Como el enfoque de este segundo sistema es hacer una recomendación en base a una película específica y no en base al usuario, como en el primer sistema, se hace una comparación en base a las recomendaciones de las películas de otros sistemas de recomendación de plataformas de streaming y se calcula su precisión y su recall.

El sistema logra obtener una precisión y recall del 80%, lo cual indica que 4 de las 5 películas recomendadas corresponden a las 5 películas recomendadas por otros sistemas de streaming.

Evaluación en base a la película “Indiana Jones and the Last Crusade”.



3. Documentación

- Objetivo

  El objetivo de este proyecto es desarrollar un sistema de recomendación de películas holístico, diseñado para ofrecer sugerencias precisas a los usuarios basándose en una película dada. Este sistema se distingue por su enfoque integral al considerar múltiples factores clave que influyen en las preferencias cinematográficas. Entre estos factores se incluyen las palabras clave asociadas a cada película, los géneros, las sinopsis, las compañías de producción, el elenco y los directores.

- Descripción del Conjunto de Datos

  El conjunto de datos contiene información sobre películas y es útil para desarrollar un programa de recomendación de películas. La información fue obtenida de TMDB y GroupLens y contiene información de películas lanzadas hasta julio del 2017.

  El dataset consta de tres archivos en formato CSV. El primer archivo es "credits.csv" e incluye los créditos de las películas y la información de los elencos y equipos de producción. El segundo archivo es “keywords.csv" y proporciona palabras clave relacionadas con las tramas de las películas. El tercer archivo es “movies.csv" y ofrece detalles adicionales sobre cada película.

  El dataset se obtuvo de Kaggle, de la siguiente liga:
  https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset?select=keywords.csv

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Secciones: Carga de Datos, Análisis del DataFrame "credits_df", Análisis del DataFrame "keywords_df", 3. Análisis del DataFrame "movies_df"):
  https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  A continuación se describen los archivos del dataset y lo que contienen.

  Credits:

  Contiene los créditos de la película y tiene la información del elenco (cast) y equipo de producción (crew).

  Contienen 3 columnas o variables:

Nombre	**Tipo de Dato**
Cast (Elenco de la película)	Object
Crew (Equipo de producción de la película)	Object
Movie ID (ID de la película)	Int64

Keywords:

Contiene palabras clave (keywords) de la trama de la película.

Contienen 2 columnas o variables:

Nombre	**Tipo de Dato**
Keywords (Palabras clave de la trama de la película)	Object
Movie ID (ID de la película)	Int64

Movies:

Contiene información adicional y detallada de la película.

Contienen 24 columnas o variables:

Nombre	**Tipo de Dato**
Adult (Película para adultos)	Object
Belongs to Collection (Pertenece a una saga la película)	Object
Budget (Presupuesto de la película)	Object
Genres (Géneros de la película)	Object
Homepage (Página web de la película)	Object
Movie ID (ID de la película)	Object
IMDB ID (Identificador único de la película)	Object
Original Language (Idioma original de la película)	Object
Original Title (Título original de la película)	Object
Overview (Sinopsis de la película)	Object
Popularity (Popularidad de la película)	Object

Poster Path (Link a poster de la película)	Object
Production Companies (Compañías productoras de la película)	Object
Production Countries (Países donde se produjo la película)	Object
Release Date (Fecha de estreno de la película)	Object
Revenue (Ganancia económica de la película)	Float64
Runtime (Duración de la película)	Float64
Spoken Languages (Idiomas en la que se encuentra disponible la película)	Object
Status (Estatus de la película cómo estrenada, en producción, etc.)	Object
Tagline (Slogan de la película)	Object
Title (Título de la película)	Object
Video (Archivo de Video)	Object
Vote Average (Calificación promedio que recibe la película)	Float64
Vote Count (Total de calificaciones que recibió la película)	Float64

- Pasos de Preprocesamiento del Conjunto de Datos

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Secciones: Nuevo Conjunto de Datos (DataFrame) Unificado, Manejo de Valores Faltantes, Formato del Conjunto de Datos (DataFrame), Conjunto de Datos (DataFrame) Listo, Análisis del Conjunto de Datos (DataFrame) Listo): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

1. Selección de variables relevantes de cada archivo.

Archivo	**Columnas o Variables**
Keywords	Movie ID, Keywords

Credits	Movie ID, Cast, Crew
Movies	Movie ID, Genres, Original Title, Overview, Production Companies

2. Convertir la variable Movie ID del archivo Movies a Int64 para que esté en el mismo formato que las variables Movie ID de los archivos Keywords y Credits.

3. Creación de un data frame unificado los tres archivos (Credits, Keywords, Movies) con las variables o columnas seleccionadas. La unión se realiza en base a la columna Movie ID de cada archivo.

4. Búsqueda de registros vacíos en las variables.

Variable	**Registros Vacíos**
Production Companies	4
Overview	995

5. Llenado por conocimiento de los registros vacíos.

   Como la variable Production Companies presenta muy pocos registros vacíos, se investigaron los registros faltantes y se llenaron sus registros.

6. Eliminación de un registro vacío

   La variable Production Companies presenta un registro vacío, sin embargo, el nombre de la película está en caracteres chinos, por lo que no se entiende el nombre de la película y no se puede hacer la investigación del registro faltante. Como solo es un registro, no hay mayor complicación en eliminarlo.

7. Los valores faltantes de la variable Overview no tienen mayor afectación en el algoritmo de recomendación por lo que se agrega una lista vacía a cada registro, ya que estas películas no tienen público su overview en el idioma en el que se está trabajando (películas sólo publicadas en China).

8. Se convierten a lista los strings de las variables Genres, Keywords y Production Companies para un mejor rendimiento del algoritmo de recomendación.

9. De la variable Crew se obtiene el string del nombre del director y se convierte a lista para un mejor rendimiento del algoritmo de recomendación.

10. De la variable Cast se obtienen los nombres de los actores o personajes (película animada) y se convierten a lista para un mejor rendimiento del algoritmo de recomendación.

11. Las palabras de la variable Overview en formato string se convierten en lista para un mejor rendimiento del algoritmo de recomendación.

12. El conjunto de datos después del preprocesamiento queda de la siguiente manera.

	id	keywords	genres	original_title	overview	production_companies	cast	crew
0	862	[jealousy, toy, boy, friendship, friends, riva...	[Animation, Comedy, Family]	Toy Story	[Led, by, Woody,, Andy's, toys, live, happily,...	[Pixar Animation Studios]	[Tom Hanks, Tim Allen, Don Rickles, Jim Varney...	[John Lasseter]
1	8844	[board game, disappearance, based on children'...	[Adventure, Fantasy, Family]	Jumanji	[When, siblings, Judy, and, Peter, discover, a...	[TriStar Pictures, Teitler Film, Interscope Co...	[Robin Williams, Jonathan Hyde, Kirsten Dunst,...	[Joe Johnston]
2	15602	[fishing, best friend, duringcreditsstinger, o...	[Romance, Comedy]	Grumpier Old Men	[A, family, wedding, reignites, the, ancient, ...	[Warner Bros., Lancaster Gate]	[Walter Matthau, Jack Lemmon, Ann-Margret, Sop...	[Howard Deutch]
3	31357	[based on novel, interracial relationship, sin...	[Comedy, Drama, Romance]	Waiting to Exhale	[Cheated, on,, mistreated, and, stepped, on,, ...	[Twentieth Century Fox Film Corporation]	[Whitney Houston, Angela Bassett, Loretta Devi...	[Forest Whitaker]
4	11862	[baby, midlife crisis, confidence, aging, daug...	[Comedy]	Father of the Bride Part II	[Just, when, George, Banks, has, recovered, fr...	[Sandollar Productions, Touchstone Pictures]	[Steve Martin, Diane Keaton, Martin Short, Kim...	[Charles Shyer]

- Exploración del Conjunto de Datos Final

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Secciones: Conjunto de Datos (DataFrame) Listo, Análisis del Conjunto de Datos (DataFrame) Listo, Gráficos): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Dimensiones:

  Número de Filas: 4662
  Número de Columnas: 8

  Variables:

  id
  keywords
  genres
  original_title
  overview
  production_companies
  cast
  crew

  Tipos de Datos:

  id (Int64)
  keywords (Object)
  genres (Object)
  original_title (Object)
  overview (Object)
  production_companies (Object)
  cast (Object)
  crew (Object)

  Porcentaje Datos Faltantes:

  id (0%)
  keywords (0%)
  genres (0%)
  original_title (0%)
  overview (0%)
  production_companies (0%)
  cast (0%)
  crew (0%)

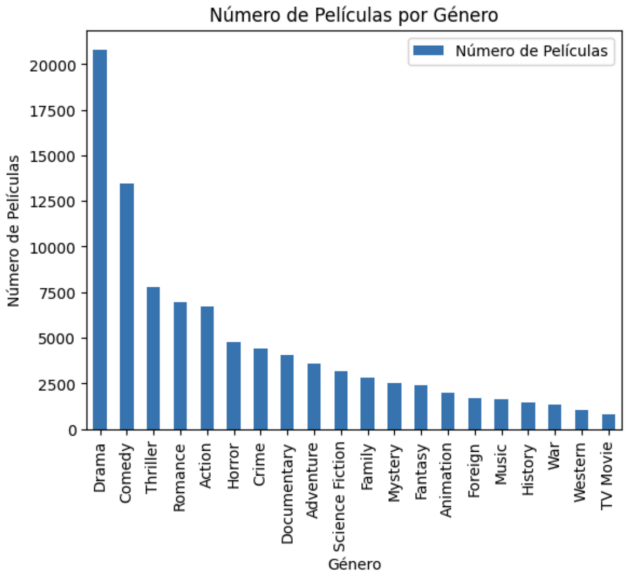
- Análisis del Conjunto de Datos Final

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Secciones: Conjunto de Datos (DataFrame) Listo, Análisis del Conjunto de Datos (DataFrame) Listo, Gráficos): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Variables:

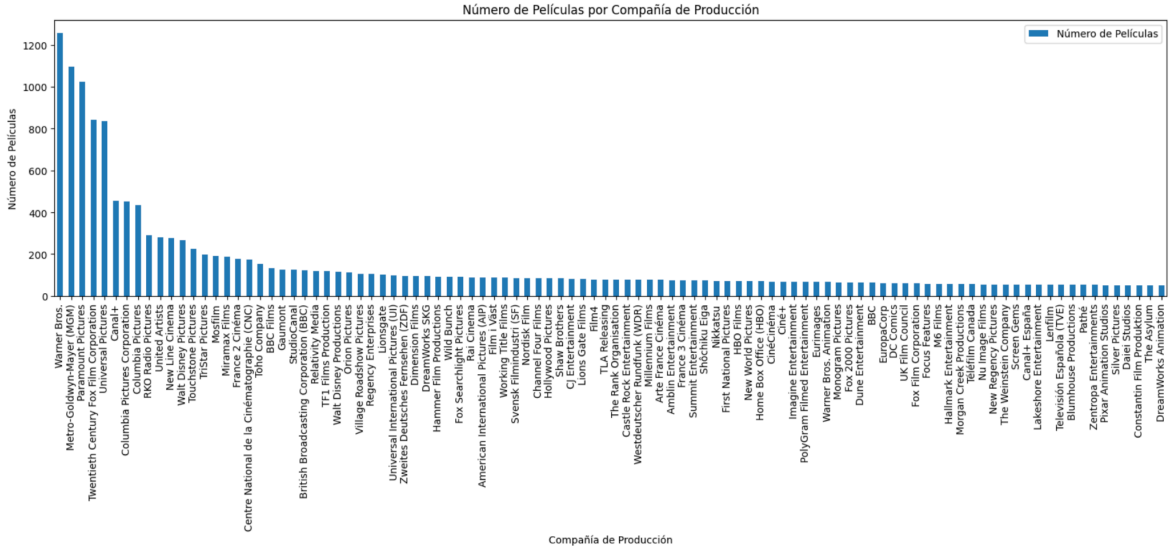
  1. genres

     Son 20 géneros de películas distintos. Se muestra una gráfica del número de películas por género.



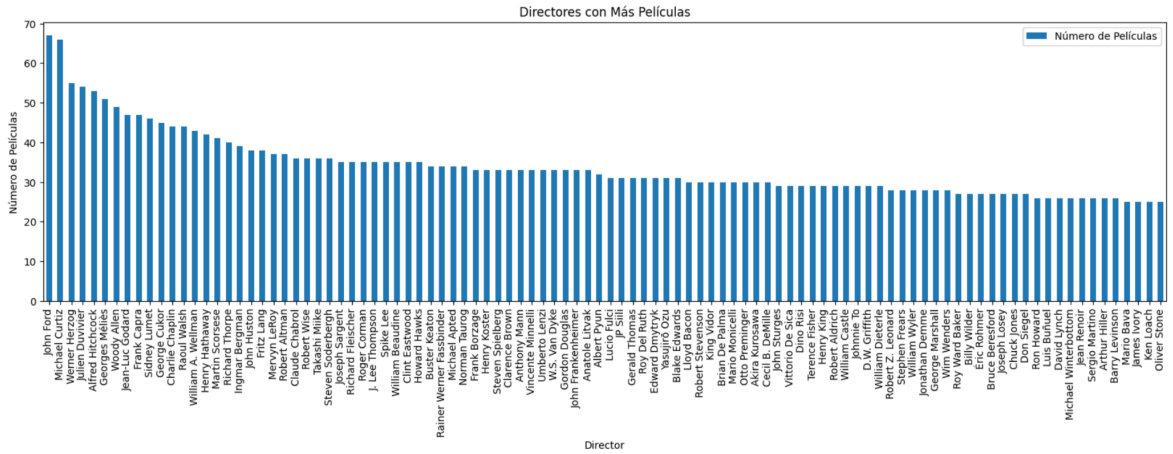
  2. production_companies

     Hay una variedad bastante grande de compañías de producción de películas. Se muestra una gráfica con las 100 compañías de producción con más películas en orden descendente.



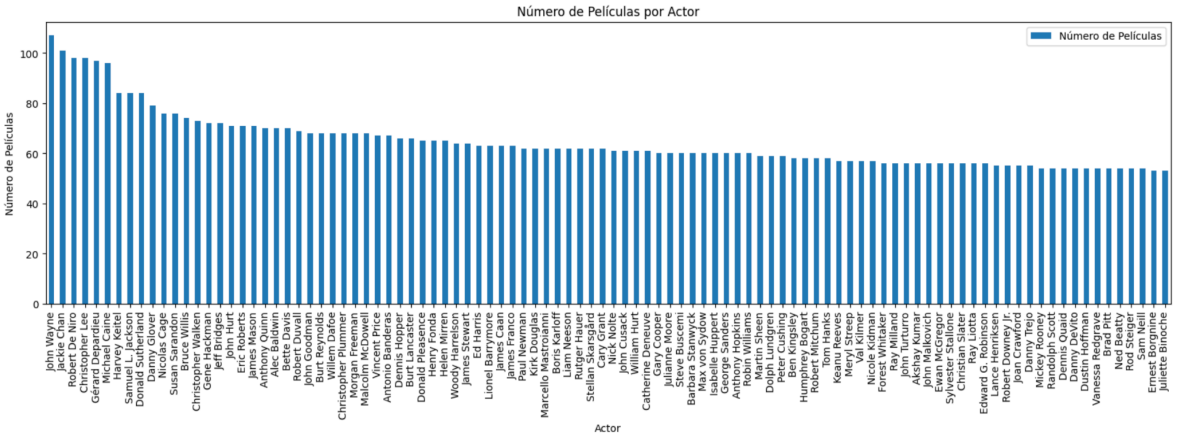
3. crew

   Hay una variedad bastante grande de directores de películas. Se muestra una gráfica con los 100 directores con más películas en orden descendente.



4. cast

   Hay una variedad bastante grande de actores y personajes (películas animadas). Se muestra una gráfica con los 100 actores o personajes (películas animadas) con más películas en orden descendente.



5. id

   Todas las películas tienen un ID distinto, por lo que se tienen 4662 valores diferentes.

6. original_title

   Todas las películas tienen títulos distintos, por lo que se tienen 4662 valores diferentes.

7. keywords

   Son palabras clave de la trama de la película, por lo que se tiene una lista con un vocabulario distinto por película.

8. overview

   Son palabras de la sinopsis de la película, por lo que se tiene una lista con un vocabulario distinto por película.

- Algoritmo de Recomendación - Red Neuronal Profunda - Autoencoder

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 1): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Este sistema de recomendación de películas utiliza una red neuronal autoencoder para hacer recomendaciones personalizadas en base a los usuarios, las películas vistas y la calificación de las películas.

  Para implementar el sistema se hace lo siguiente:

  1. Creación de la Tabla Pivote: Se crea una tabla pivote donde cada fila representa a un usuario y cada columna representa una película, con las calificaciones dadas por los usuarios. Las calificaciones faltantes se llenan con ceros.

  2. Normalización de Datos: Se normalizan las calificaciones para que estén en un rango de 0 a 1, lo cual mejora la eficiencia del entrenamiento del modelo.

  3. División de Datos: Se dividen los datos en conjuntos de entrenamiento y prueba, usando el 80% de los datos para entrenamiento y el 20% para prueba.

  4. Definición y Construcción del Modelo: Se define y construye un modelo de red neuronal (autoencoder) con:

     - Una capa de entrada que toma como entrada el número de películas.

     - Una capa oculta densa con 256 nodos y función de activación sigmoid.

     - Una capa de salida con un nodo por cada película.

     - El modelo se compila usando el optimizador Adam y la pérdida se mide con el error cuadrático medio (MSE). También se añaden las métricas adicionales MAE y MAPE.

  5. Entrenamiento del Modelo: Se entrena el modelo usando los datos de entrenamiento, configurado para 50 épocas y un tamaño de lote de 64. Dado que se trata de un autoencoder, tanto la entrada como el objetivo son los mismos datos.

  6. Evaluación del Modelo: Se evalúa el modelo en el conjunto de prueba, imprimiendo la pérdida y las métricas adicionales (MAE, MSE, MAPE).

  7. Cálculo del RMSE: Se calcula el Root Mean Squared Error (RMSE) para tener una métrica adicional de evaluación del modelo.

  8. Visualización de la Historia de Entrenamiento: Se grafica la pérdida de entrenamiento y validación a lo largo de las épocas para visualizar el desempeño del modelo durante el entrenamiento.

  10. Al final, se realiza una prueba dando como input el número de un usuario X y se obtiene el top 10 de recomendaciones de películas.

Como referencia se utilizó el usuario con el ID número “10”, dónde el algoritmo devolvió las respectivas recomendaciones acorde a las características.

```
#Intentar con diferentes user_ids para obtener las peliculas recomendadas
get_Recommendations_for_user(10)
```

```
1/1 [==============================] - 0s 87ms/step
Top 10 movie recommendations for the sample user:
title: Secret of Roan Inish, The (1994), Predicted Rating: 0.5054865479469299
title: Hate (Haine, La) (1995), Predicted Rating: 0.44272491335868835
title: Miracle on 34th Street (1994), Predicted Rating: 0.4089670777320862
title: Just Cause (1995), Predicted Rating: 0.38791871070861816
title: Lost Weekend, The (1945), Predicted Rating: 0.37184029817581177
title: Poetic Justice (1993), Predicted Rating: 0.3688317537307739
title: Singin' in the Rain (1952), Predicted Rating: 0.3478822410106659
title: Go Fish (1994), Predicted Rating: 0.3141983151435852
title: Some Like It Hot (1959), Predicted Rating: 0.3121604025363922
title: Being Human (1993), Predicted Rating: 0.3114105463027954
```

- Algoritmo de Recomendación - NLP y Similitud Coseno

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 2): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Este sistema de recomendación de películas utiliza las palabras clave, el elenco, el director, los géneros, el título, la sinopsis y la compañía de producción como características para calcular la similitud entre películas y recomendar películas similares a una película de entrada específica y hacer recomendaciones personalizadas.

  Este sistema de recomendación de películas implementa técnicas de procesamiento de texto, análisis de similitud y modelado de datos.

  Para implementar el sistema se hace lo siguiente:

  1. Creación de una Colección Unificada de Características

  Al conjunto de datos se agrega una columna adicional llamada “collection” en la que se concatenan todas las características a tomar en cuenta (palabras clave, elenco, director, géneros, título, sinópsis y compañía de producción) en una sola cadena de texto por película.

  Esta colección unificada facilita el procesamiento, ya que consolida toda la información relevante de una película en un solo campo.

  2. Transformación de Texto

  Se eliminan los espacios en blanco y se convierte todo a minúsculas.

  Se aplica el algoritmo de stemming, “PorterStemmer”, de la librería “nltk”, que reduce las palabras a sus raíces morfológicas, lo cual es crucial para manejar las variantes de las palabras y mejorar la coincidencia de términos.

  3. Vectorización de Texto

  Se utiliza “CountVectorizer” de la librería “scikit-learn” para convertir las cadenas de texto en una representación numérica que sea más fácil de procesar. Se configuran con un máximo de 5000 características y se eliminan las “stop words”.

  La salida de “CountVectorizer” es una matriz dispersa que representa la frecuencia de cada término en el corpus.

4. Cálculo de Similitud - Similitud Coseno

La recomendación de sistema por similitud de coseno es un enfoque utilizado en sistemas de recomendación para calcular la similitud entre dos elementos basándose en sus características.

El cálculo de la similitud de coseno se basa en el concepto de espacio vectorial, donde cada elemento se representa como un vector en un espacio multidimensional, donde cada dimensión corresponde a una característica.

El algoritmo de similitud de coseno mide el ángulo entre dos vectores, lo que refleja la similitud direccional entre ellos. Cuanto más cercano estén los vectores en dirección (es decir, cuanto menor sea el ángulo entre ellos), mayor será la similitud de coseno y, por lo tanto, mayor será la similitud entre los elementos.

Para calcular la similitud de coseno entre dos elementos, se utiliza la siguiente fórmula:

$$\text{similitud}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \cdot \|\mathbf{B}\|}$$

Donde:

- $\mathbf{A} \cdot \mathbf{B}$ es el producto punto entre los vectores $\mathbf{A}$ y $\mathbf{B}$.
- $\|\mathbf{A}\|$ y $\|\mathbf{B}\|$ son las magnitudes (normas) de los vectores $\mathbf{A}$ y $\mathbf{B}$ respectivamente.

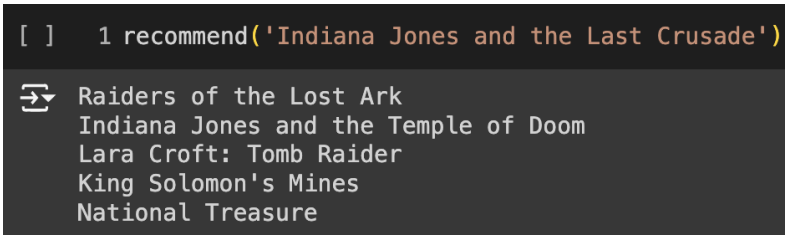
En este contexto, la similitud de coseno ayuda a determinar qué tan similares son las películas en función de sus descripciones textuales vectorizadas.

5. Algoritmo de Recomendación

Se implementa una función de recomendación dónde:

- Se toma el título de una película como entrada y se busca su índice en el DataFrame.
- Se calculan las distancias de similitud coseno entre la película dada y todas las demás películas.
- Se ordenan las distancias en orden descendente y se seleccionan las cinco películas más similares.
- Se muestran los títulos de las películas recomendadas.

Como referencia se utilizó la película “Indiana Jones and the Last Crusade”, dónde el algoritmo devolvió las respectivas recomendaciones acorde a las características.



- Evaluación Algoritmo de Recomendación - Red Neuronal Profunda - Autoencoder

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 2): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  Las métricas a utilizar para evaluar el desempeño de este sistema son las siguientes:

RMSE (Root Mean Squared Error) - Error Cuadrático Medio de la Raíz:

- El RMSE mide la diferencia entre los valores predichos por el modelo y los valores reales de las calificaciones. Es útil para evaluar la precisión de las predicciones.
- Fórmula: $\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$

  donde y_i son las calificaciones reales y $\hat{y}_i$ son las calificaciones predichas.

MAE (Mean Absolute Error) - Error Absoluto Medio:

- El MAE mide la magnitud promedio de los errores en un conjunto de predicciones, sin considerar su dirección. Es más interpretativo en términos de unidades de las calificaciones.
- Fórmula: $\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$

MAPE (Mean Absolute Percentage Error)

- **Definición:** Mide la precisión de un modelo de predicción en términos porcentuales.
- **Interpretación:** Proporciona una idea clara de la precisión relativa del modelo comparada con los valores reales, expresada como un porcentaje.
- **Ventajas:**
  - Independiente de la escala de los datos.
  - Fácil de interpretar.
- **Desventajas:**
  - Sensible a valores reales cercanos a cero, lo que puede causar valores muy altos o indefinidos.

Fórmula del MAPE

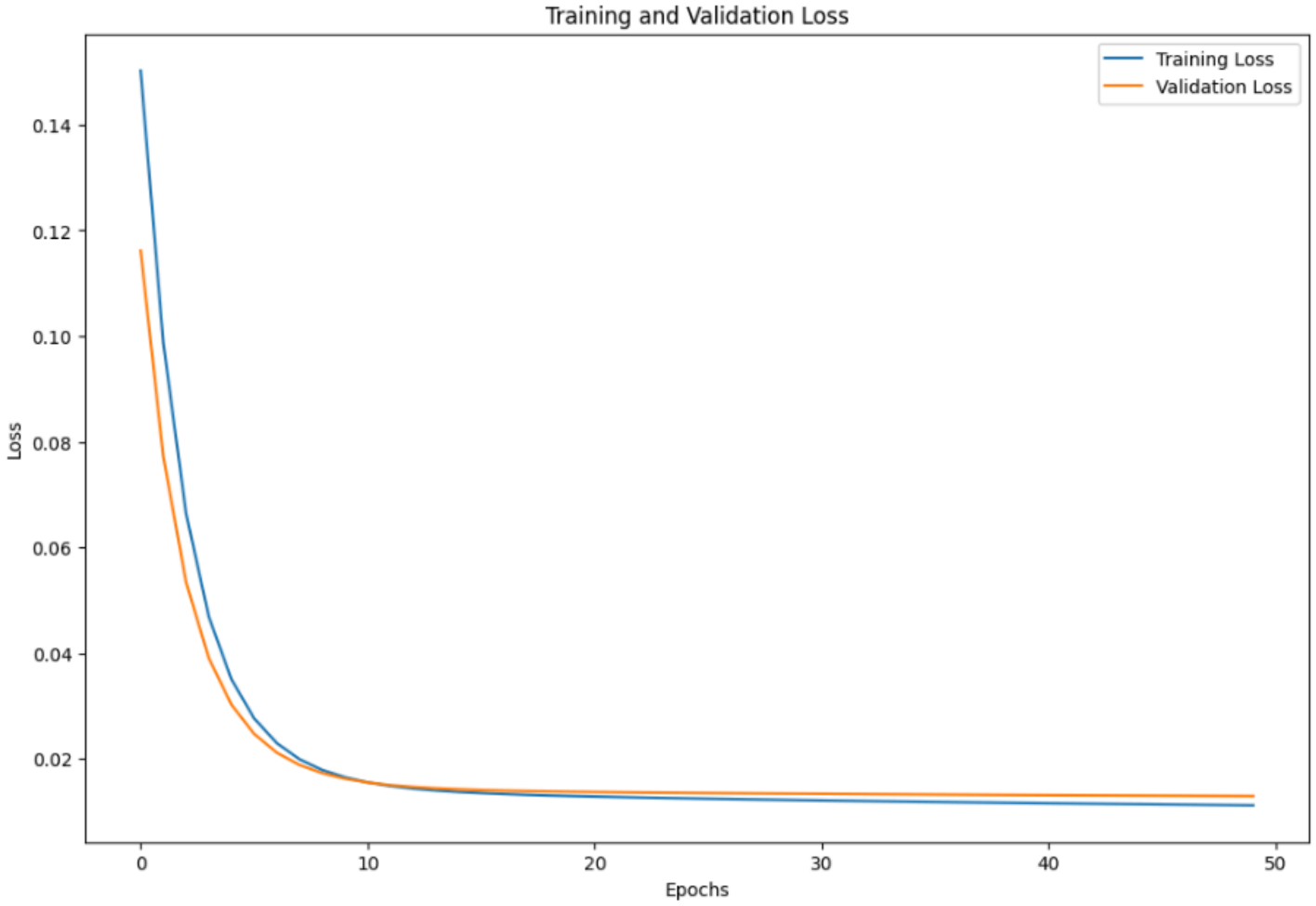
$\text{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$

donde:

- y_i son las calificaciones reales.
- $\hat{y}_i$ son las calificaciones predichas.
- n es el número de observaciones.

Resultados con 50 epocas:

```
Root Mean Squared Error (RMSE): 0.11379894263401093
4/4 - 0s - loss: 0.0130 - mae: 0.0594 - mse: 0.0130 - mape: 47885120.0000 - 48ms/epoch - 12ms/step
Loss: 0.012950199656188488
Mean Absolute Error (MAE): 0.05944839119911194
Mean Squared Error (MSE): 0.012950199656188488
Mean Absolute Percentage Error (MAPE): 47885120.0
```



- Evaluación Sistema de Recomendación 2

  Nota: El proceso y código descrito a continuación se puede ver a detalle en la siguiente liga de Google Colab (Sección: Sistema de Recomendación 2): https://colab.research.google.com/drive/10bgFhfKPnCFh5lZiEVtTzFhz6fgkUjxo?usp=sharing

  1. Evaluación con Base a la Similitud de Coseno

  En este sistema de recomendación basado en la similitud por coseno, donde se utilizan características de palabras clave asociadas a cada película como los géneros, las sinopsis, las compañías de producción, el elenco y los directores, un valor de similitud mayor a 0.20 representa un muy buen desempeño debido a la naturaleza dispersa y multidimensional de los datos. En este contexto, las características que describen cada película son numerosas y variadas, lo que resulta en vectores de alta dimensionalidad. Dado que la similitud por coseno mide el ángulo entre estos vectores, un valor superior a 0.20 indica que dos películas tienen un grado considerable de similitud en varios de estos aspectos, a pesar de la diversidad de datos. Esto es especialmente significativo porque, en un espacio vectorial tan amplio, la mayoría de los pares de películas tienden a tener valores de similitud bajos, cercanos a 0. Por tanto, alcanzar un valor superior a 0.20 sugiere que las películas comparten suficientes características relevantes para que la recomendación sea efectiva y precisa.

En el sistema se logra obtener un valor mayor a 0.20, en las 5 recomendaciones que se hacen, lo cual índica que el sistema tiene un buen desempeño.

2. Evaluación de Correlación de las Películas Recomendadas en Base a su Similitud de Coseno.

De igual manera, se logra obtener una correlación mayor a 0.20 en la similitud de coseno entre las 5 películas recomendadas. Esto indica que el sistema tiene un buen desempeño.

3. Evaluación de Precisión y Recall en Base a Otros Sistemas de Recomendación.

Como el enfoque de este segundo sistema es hacer una recomendación en base a una película específica y no en base al usuario, como en el primer sistema, se hace una comparación en base a las recomendaciones de las películas de otros sistemas de recomendación de plataformas de streaming y se calcula su precisión y su recall.

El sistema logra obtener una precisión y recall del 80%, lo cual indica que 4 de las 5 películas recomendadas corresponden a las 5 películas recomendadas por otros sistemas de streaming.

Evaluación en base a la película “Indiana Jones and the Last Crusade”.